Assignment 7: Tyler McCallum & Pablo Landa

Task 1: Running the Neural Net with xor data

```
nn = NeuralNet(2, 5, 1)
nn.train(xor_data)

print(nn.evaluate([0.0, 1.0]))

for triple in nn.test_with_expected(xor_data):
   print(triple)

for i in nn.test_with_expected(xor_data):
   print(f"desired: {i[1]}, actual: {i[2]}")
```

Error after 100 iterations: 0.44601201785739764
Error after 200 iterations: 0.2433681255871073
Error after 300 iterations: 0.08306067812732068
Error after 400 iterations: 0.037467574167650335
Error after 500 iterations: 0.022074966910072817
Error after 600 iterations: 0.01507101567507406
Error after 700 iterations: 0.011225422781205599
Error after 800 iterations: 0.008844878246726905
Error after 900 iterations: 0.007245875627676437
Error after 1000 iterations: 0.006106893364831538
[0.9384785592796963]
([0.0, 0.0], [0.0], [0.04606542275386509])
([0.0, 1.0], [1.0], [0.9384785592796963])
([1.0, 0.0], [1.0], [0.9509829459981699])
([1.0, 1.0], [0.0], [0.061674274423464605])
desired: [0.0], actual: [0.04606542275386509]
desired: [1.0], actual: [0.9384785592796963]
desired: [1.0], actual: [0.9509829459981699]
desired: [0.0], actual: [0.061674274423464605]

Task 2: Running with a single hidden node:

```
from neural import NeuralNet

# each row is an (input, output) tuple
xor_data = [
   # input output corresponding example
   ([0.0, 0.0], [0.0]), #[0, 0] => 0
   ([0.0, 1.0], [1.0]), #[0, 1] => 1
   ([1.0, 0.0], [1.0]), #[1, 1] => 1
   ([1.0, 1.0], [0.0])  #[1, 0] => 0
]
nn = NeuralNet(2, 1, 1)
nn.train(xor_data)

print(nn.evaluate([0.0, 1.0]))

for triple in nn.test_with_expected(xor_data):
   print(triple)

for i in nn.test_with_expected(xor_data):
   print(f"desired: {i[1]}, actual: {i[2]}")
```

```
pablolandacatan@dhcp-10-105-201-80 Assignment 7 % /usr/local/bin/python3
"/Users/pablolandacatan/Desktop/Kellogg/MBAi 410/Assignment 7/Assignment7/main.py"
Error after 100 iterations: 0.5183319662173628
Error after 200 iterations: 0.5082052157711499
Error after 300 iterations: 0.47461674492738803
Error after 400 iterations: 0.4163483872609569
Error after 500 iterations: 0.38238496260119925
Error after 600 iterations: 0.3679661400662856
Error after 700 iterations: 0.3608257517334754
Error after 800 iterations: 0.3566977876507337
Error after 900 iterations: 0.3540425664200413
Error after 1000 iterations: 0.35220345050666274
[0.6585229669788363]
([0.0, 0.0], [0.0], [0.07787083099822222])
([0.0, 1.0], [1.0], [0.6585229669788363])
([1.0, 0.0], [1.0], [0.6580785828372354])
([1.0, 1.0], [0.0], [0.6674944898890532])
desired: [0.0], actual: [0.07787083099822222]
desired: [1.0], actual: [0.6585229669788363]
desired: [1.0], actual: [0.6580785828372354]
```

desired: [0.0], actual: [0.6674944898890532]

Task 3: Developing a Neural Net that will run on the Wine Test data (see main.py):

```
/Users/tylermccallum/Projects/mbai410/week7
% python main.py
Loading wine dataset...
Mosts: [14.83, 5.8, 3.23, 30.0, 162.0, 3.88, 5.08, 0.66, 3.58, 13.0, 1.71, 4.0, 1680.0]
Leasts [11.03, 0.74, 1.36, 10.6, 70.0, 0.98, 0.34, 0.13, 0.41, 1.28, 0.48, 1.27, 278.0]

Loaded 142 training samples and 36 test samples.
Training neural network...
Error after 1000 iterations: 0.058200811025246935
Testing neural network
Accuracy: 0.8888888888888888
Predictions
Example 1: Actual=1, Predicted=1, Output=[0.003]
Example 2: Actual=1, Predicted=1, Output=[0.005]
Example 3: Actual=1, Predicted=1, Output=[0.788]
Example 4: Actual=1, Predicted=1, Output=[0.508]
Example 5: Actual=1, Predicted=1, Output=[0.403]
```